키넥트를 이용한 중첩이 있는 제스처의 효율적 인식

조성원 O1 장하영 2 장병탁 2

한국과학영재학교 1 서울대학교 전기컴퓨터공학부 2

a1sams1a@naver.com, hyjang@bi.snu.ac.kr, btzhang@bi.snu.ac.kr

Effective Recognition of Overlapped Gesture using Kinect

Sungwon Cho O1 Ha-Young Jang 2 Byoung-Tak Zhang 2

Korea Science Academy 1

School of Computer Science and Engineering, Seoul National University 2

요 약

위치 정보를 이용한 특징기반 제스처 인식은 단순한 알고리즘으로도 빠르게 처리할 수 있다는 장점이 있지만, 주변 환경에 민감하고 특징점의 중첩이 있을 경우에 처리가 어렵다는 단점이 있다. 본 논문에서는 특징점간의 중첩이 발생하는 제스처의 효율적 인식을 위한 포스처 및 시간 정보의 활용법을 제안한다. 원, 삼각형, 사각형 등의 특징점이 중첩되는 제스처들의 경우에는 위치 정보만을 이용한 특징기반 제스처 인식에 어려움이 있지만 제안한 방법론을 이용하여 보다 좋은 결과를 얻어낼 수 있다. 특히 시간 정보의 활용은 제스처 인식뿐 아니라 동작 추적 및 모션 생성 등의 다양한 분야에 적용할 수 있는 가능성을 가지고 있다.

1. 서론

사용자와 기계의 소통 방법은 천공카드로부터 마우스나 키보드, 멀티 터치까지 계속하여 발전해 왔고, 사용자가 기계와 더 편하게 의사소통을 할 수 있도록 많은 연구를 통해 개선되거나 새로운 소통 방법이 연구되고 있다. 이 중 제스처 인식은 신체 모두를 인식하여 처리하기에는 기술적 한계가 있어 제약 조건에서만 사용한다. 그러나 이는 프레젠테이션 조작[1] 이나 비디오 게임 조작[4] 같은 새로운 사용자 인터페이스[5]를 설계하기에 매우 편할 뿐만 아니라 마우스나 키보드 등의 새로운 장비를 사용자가 익힐 필요가 없다는 점에서 사용자가 쉽게 쓸 수 있다. 제스처를 인식하기 위해서는 사용자의 움직임을 먼저 인식하고 데이터를 분석해야 한다. 이를 위해서는 비디오 영상 등을 이용하여 직접 영상처리[2, 3]를 하거나 비디오 영상과 함께 적외선 센서를 사용한 Microsoft Kinect 장비의 등의 API를 사용[7]하여 처리한다. 또한 데이터를 분석하는 방법에는 순환 신경망[6] 등의 알고리즘을 사용한다. 원시 데이터를 분석할 수 있는 데이터로 만드는 방법은 여러 종류가 있다. 대표적으로 위치 정보와 HMM(Hidden Markov Model)를 이용하여 특징점을 추출해 내는 방법이 있으나, 이는 주변 환경과 데이터의 오류에 민감하고 특징점이 중첩되는 경우에는 인식하기 힘들다. 본 논문에서는 이러한 특징점이 중첩되는 체스처를 Microsoft Kinect와 다중 퍼셉트론 (Multilayer Perceptron) 알고리즘을 사용하여 인식할 때에 인식률을 높이기 위한 방법을 제안하였다.

2. 중첩이 있는 제스처의 분류

제스처의 데이터는 보통 시간에 따른 좌표 값으로 이루어진다. 이러한 좌표 값의 변화를 기계학습 알고리즘으로 분류한 결과를 이용하여 제스처를 인식하게 된다. 한 평면에서 구성되는 'ㄱ'이나 'ㄴ'자 같은 제스처는 시간에 따른 좌표 이동이 서로 명확하게 구분 될 수 있어 인식하기가 상대적으로 쉽다. 반면에, 원, 삼각형, 사각형 등의 시간에 따른 좌표 이동이 명확하게 구별되지 않는 제스처는 상대적으로 인식하기가 어렵다. 본 논문에서는 위의 3가지 제스처를 분류하는 실험을 수행하였다. 3가지 제스처 모두 오른손이 시계방향으로 움직이며 원이나 삼각형의 경우 상단 중앙, 사각형의 경우에는 상단 왼쪽에서 시작하는 제스처로 정의하였다.

데이터 수집은 Microsoft Kinect의 API인 Skeleton Model에서 HAND_RIGHT의 Joint 데이터의 x, y 값을 사용하였고, 각 제스처마다 80번을 반복 시행하여 데이터를 수집하였다. 원시 데이터의 경우 각각의 프레임은 시간과 오른손의 좌표로 나타내어 진다. i번째 프레임 x_i의 데이터를 아래와 같이 정의할 때,

$$x_i =< t, R_x, R_y > \quad (1)$$

t는 지역 시간을, R_x는 오른손의 x좌표를 R_y는 오른손의 y좌표를 각각 나타난다.

이 때, v_i는 시간을 제외한 데이터 x_i과 x_{i-1}과의 차이로 정의된다.

$$v_i =< c_x, c_y >=< R_{i,x} - R_{i-1,x}, R_{i,y} - R_{i-1,y} > \quad (2)$$

$R_{i,x}$, $R_{i,y}$ 는 각각 i번째 프레임의 오른손의 x 좌표와 y좌표를, $R_{i-1,x}$, $R_{i-1,y}$ 는 각각 i-1번째 프레임의 오른손의 x 좌표와 y좌표를 나타낸다.

이 때, 제스처 G는 벡터 v_i의 순서가 있는 집합으로 표현되고,

$$G = v_1 v_2 v_3 \cdots\cdots v_n \quad (3)$$

n은 한 제스처의 길이를 나타낸다.

제스처를 수행하는데 걸리는 시간, 제스처의 크기, 제스처의 모양 등이 다르기 때문에, 동일한 제스처에 대한 데이터의 분포에도 분산이 커지기 때문에, 이로 인해 인식률이 떨어지는 문제를 해결하기 위해 다음과 같이 원시데이터에 대한 전처리를 수행한다..

먼저 사람마다 한 제스처를 할 때 도형을 그리는 시간의 차이를 정규화하기 위하여 프레임의 수 n을 고정된 값으로 변경한다. n이 너무 작을 경우 제스처를 제대로 표현할 수 없고 n이 너무 클 경우 feature의 개수가 증가해 제스처를 분류하는 시간이 증가한다. 이를 위해 프레임의 수 n을 다른 프레임의 수 k로 변경하는 공식은 아래와 같다. $(k \leq \frac{n}{2})$

$$G = v_{\frac{n}{k}\times 0+1} v_{\frac{n}{k}\times 1+1} v_{\frac{n}{k}\times 2+1} \cdots\cdots v_{\frac{n}{k}\times (k-1)+1} \quad (4)$$

또한 제스처를 수행할 때마다 그리는 원, 삼각형, 사각형의 크기는 달라질 수 밖에 없고 이에 따라 i번째 프레임 벡터 v_i의 크기는 달라진다. 제스처 G의 모든 벡터 $v_1 \sim v_n$의 크기는 아래 수식을 이용하여 1로 정규화된다.

$$v_i = < \frac{c_x}{\|v_i\|}, \frac{c_y}{\|v_i\|} > \quad (5)$$

이 연구에서 사용된 원, 삼각형, 사각형 제스처의 특징은 평면상에서 특징점이 겹치는 부분이 많이 존재한다는 것이다. 그러나 원, 삼각형, 사각형의 기하학적 성질을 볼 때, 곡률이 변화하는 횟수가 다르다는 것을 알 수 있다. 이로 인해 위의 3가지 제스처는 각 프레임마다 곡률의 차이가 존재한다. 원의 경우 모든 프레임에서 곡률이 대체로 일정하고 삼각형, 사각형의 경우 곡률이 변하는 부분이 각각 2번, 3번 생긴다. 이를 이용하여 비선형적인 계산으로 수행된 곡률 값을 벡터 v_i에 추가였다. 곡률을 구하는 식은 아래와 같다.

$$v_i = \frac{angle(v_i - v_{i-1}, v_{i+1} - v_i)}{\|v_i - v_{i-1}\| + \|v_{i+1} - v_i\|} \quad (6)$$

이 때, $angle$은 두 벡터 사이의 각도를 나타낸다.

3. 실험 및 결과

제스처 데이터는 Microsoft Kinect를 사용하여 피실험자 1명이 각 제스처마다 80회를 반복하여 원시데이터를 수집하였다.

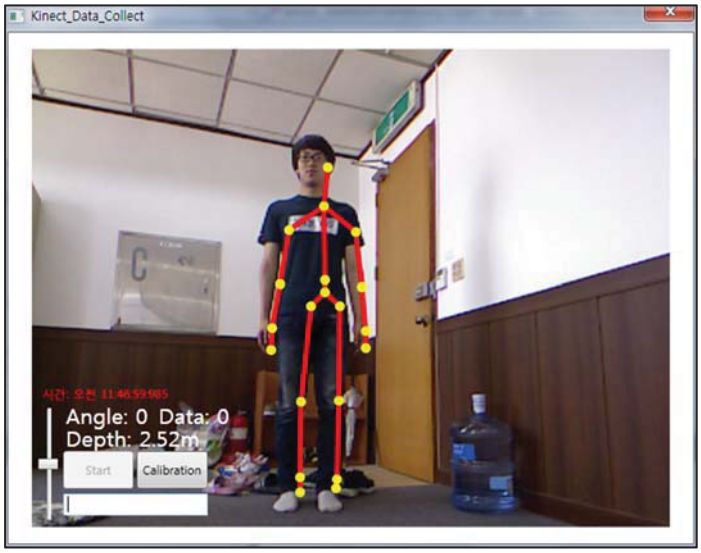

그림 1. 제스처 데이터를 수집하기 위한 프로그램의 실행 모습

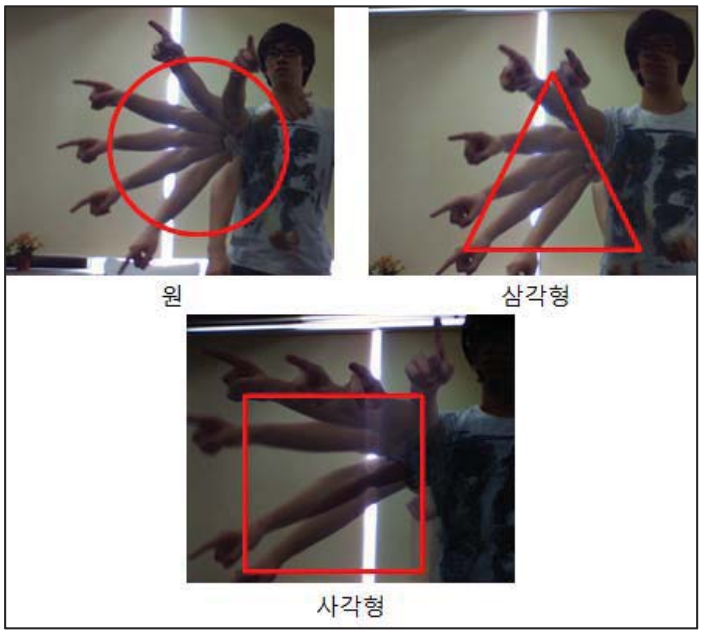

그림 2. 원, 삼각형, 사각형의 제스처를 실제로 수행하는 모습

이와 같이 수집된 원시데이터는 수식 2, 5를 사용하여 정규화된 후에 수식 4를 사용하여 프레임의 수를 고정된 값 k으로 변경하여 보정하게 된다. 그림 3은 프레임의 수를 12, 24로 하였을 때의 결과를 시각적으로 보여주고 있다.

이 외에도 프레임의 수를 프레임의 수 k을 6, 12, 18, 24, 32로 해보았으나 6프레임의 경우 인간조차 제대로 식별할 수 없어 제외하였다. 또한 대부분의 원시데이터의 전체 프레임 수가 60 미만으로 프레임의 수를 32로 할 경우 시간 간격이 균등하게 겹치지 않는 데이터를 얻을 수 없어 제외하였다.

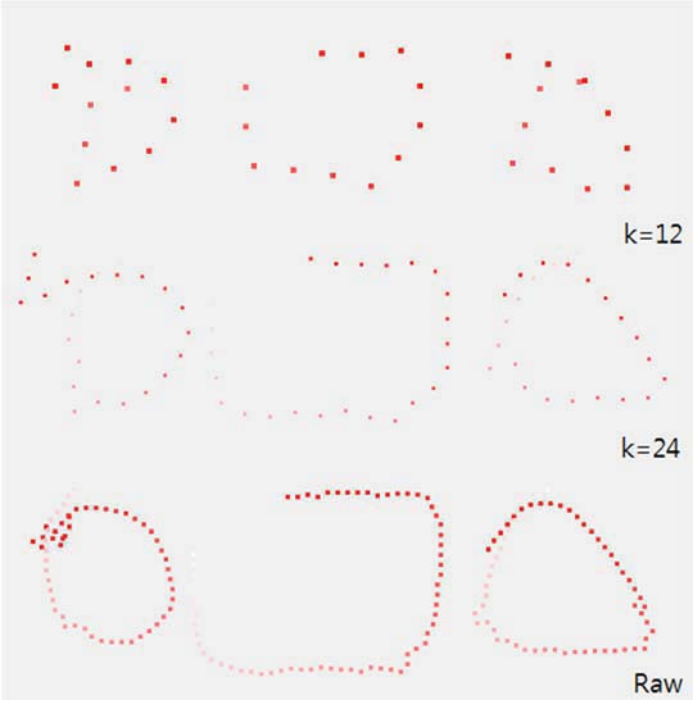

그림 3. 원본 데이터와 프레임의 수를 12, 24로 변환하였을 때 원, 사각형, 삼각형 데이터의 예시

수식 6을 이용하여 계산한 곡률 정보의 추가에 따른 성능의 차이를 확인하기 위하여 프레임의 수가 12, 18, 24인 6가지 경우에 대하여 Weka[8]에서 제공하는 다중 퍼셉트론 (Multilayer Perceptron) 학습 알고리즘을 이용하여 실험을 수행하였다. Hidden Layer의 개수를 2로 하고 Learning Rate = 0.3, Momentum = 0.2, Training Time = 500ms으로 주어 분석하였고 그 결과는 다음과 같다.

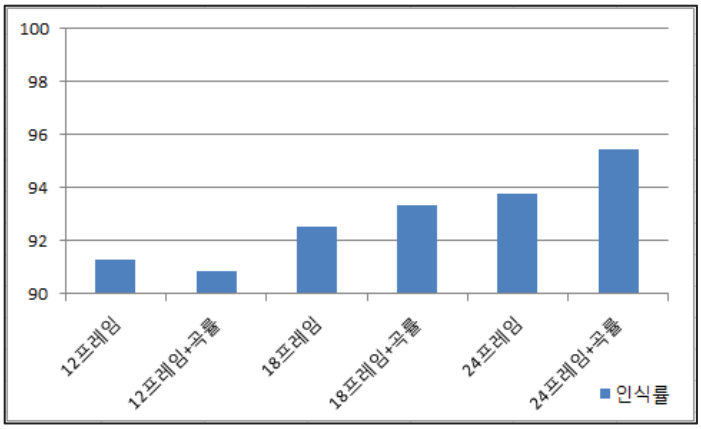

그림 4. 프레임 수와 곡률 데이터 추가에 따른 인식률

프레임의 수가 증가할수록 인식률이 높아졌으며 일반적으로 곡률을 추가했을 때 약 2~3%정도 인식률이 높아졌다. 12프레임으로 실험하였을 때는 데이터 수 자체가 적어 특징점이 겹치는 데이터가 많아 곡률을 추가한다고 해도 인식률이 증가하지 않았던 것으로 판단된다. 그 외에 18, 24프레임으로 실험하였을 때는 추가된 곡률 데이터가 각 제스처의 특징을 반영하여 인식률이 높아졌다고 판단된다.

4. 결론 및 향후 연구방향

본 연구에서 수행한 제스처 인식은 현재 사용되고 있는 마우스나 키보드, 터치 스크린 등의 인터페이스 장비가 없이도 컴퓨터나 TV 등 다른 전자 제품과의 인터페이스에 적용될 수 있다. 본 연구에서는 여러 사람이 이용하는 이러한 제스처 인식의 정확도를 높이기 위해 제스처의 특성을 이용하여 짧은 시간 안에 높은 정확도로 인식하도록 하는 것에 대한 가능성을 보여 주었다. 그러나 많은 종류의 제스처가 있을 경우 특징점이 겹치는 부분이 증가하여 이를 인식하기 위해 입력 데이터의 개수가 증가해야 할 것으로 예상된다. 이로 인해 많은 종류의 제스처가 있을 경우 이를 인식하는데 시간이 더 길어질 수 있을 것으로 판단된다. 또한 정밀한 제스처의 경우 측정오차로 인해 정확도가 떨어질 것이라고 예상된다. 따라서, 앞으로는 많은 제스처를 빠른 시간 내에 분류할 수 있도록 feature의 개수를 줄이는 것과 오차가 있는 관측 값으로부터 실제 값을 예측해 낼 수 있는 베이즈필터링(Bayesian filtering) 등에 대한 연구가 수행되어야 할 것이다.

감사의 글

이 논문은 교육과학기술부의 재원으로 국가연구재단의 지원을 받아 수행된 연구(No. 2012-0005643)이며, KAIST 부설 한국과학영재학교 R&E의 성과물로 2012년도 한국과학창의재단의 지원을 받아 수행되었음.

참고문헌

[1] 강선미, “제스처 인식과 센서를 이용한 프레젠테이션 제어 시스템”, *한국지능시스템학회 논문지*, 제 21권, 제 4호, pp. 481-486, 2011

[2] 민병우, 윤호섭, 소정, Toshiaki Ejima, “시공간상의 궤적 분석에 의한 제스쳐 인식”, *정보과학회논문지(B)*, 제 26권, 제 1호, pp. 157-166, 1991

[3] 허승주, 이성환, “연속적인 손 제스처의 실시간 인식을 위한 계층적 베이지안 네트워크”, *정보과학회논문지 : 소프트웨어 및 응용*, 제 36권, 제 12호, pp. 1028-1033, 2009

[4] 홍동표, 우운택, “제스처기반 사용자 인터페이스에 대한 연구 동향”, *Telecommunications Review*, 제 18권, 제 3호, pp. 403-412, 2008

[5] Kang, H. and Woo Lee, C. and Jung, K, “Recognition-based gesture spotting in video games”, *Pattern Recognition Letters*, Vol. 25, Issue 15, pp. 1701-1714, 2004

[6] Murakami, K. and Taguchi, H., “Gesture recognition using recurrent neural networks”, *CHIACM*, pp. 237-242, 1991

[7] Ren, Z. and Meng, J. and Yuan, J. and Zhang, Z., “Robust hand gesture recognition with kinect sensor”, *ACM Multimedia*, pp. 759-760, 2011

[8] The University of Waikato, Weka, http://www.cs.waikato.ac.nz/ml/weka/